12

Vision-Based Planning and Execution of Precision Grasps

O. Fuentes, H.F. Marengoni, and R.C. Nelson

Technical Report 546
December 1994

UNIVERSITY OF
ROCHESTER
COMPUTER SCIENCE

Vision-based Planning and Execution of Precision Grasps

Olac Fuentes Hilcea F. Marengoni Randal C. Nelson

The University of Rochester
Computer Science Department
Rochester, New York 14627

Technical Report 546

December 1994

Abstract

In this paper we present a system for vision-based planning and execution of fingertip grasps using a four-fingered dextrous hand. Our system does not rely on prior models of the objects to be grasped; it obtains all the information it needs from vision and from tactile sensors located at the fingertips of the hand. The grasp planner is based on a genetic algorithm modified to allow the use of real numbers as the basic representation unit. The grasp executer is based on differential visual feedback, which allows the system to specify goals and monitor progress in image space without needing absolute calibration between the camera and the hand. We present experimental results showing the application of the system to grasping unknown objects with the Utah/MIT hand.

This material is based upon work supported by DARPA under research grant no. MDA972-92-J-1012, ONR under research grant no. N00014-93-1-0221, and Brazilian CNPq/RHAE under grant no. 610330/92-6. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA, ONR or the Brazilian CNPq/RHAE.

REPORT DOCUMENTATION PAGE

Form Approved

OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)

2. REPORT DATE
December 1994

3. REPORT TYPE AND DATES COVERED
technical report

4. TITLE AND SUBTITLE
Vision-Based Planning and Execution of Precision Grasps

5. FUNDING NUMBERS
N00014-93-1-0221, MDA972-92-J-1012

6. AUTHOR(S)
O. Fuentes, H.F. Marengoni, and R.C. Nelson

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES
Computer Science Dept.
734 Computer Studies Bldg.
University of Rochester
Rochester NY 14627-0226

8. PERFORMING ORGANIZATION

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESSES(ES)

Office of Naval Research	ARPA
Information Systems	3701 N. Fairfax Drive
Arlington VA 22217	Arlington VA 22203

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
TR 546

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Distribution of this document is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)
(see title page)

14. SUBJECT TERMS
grasping; genetic algorithm; visual servoing

15. NUMBER OF PAGES
22 pages

16. PRICE CODE
free to sponsors; else $2.00

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
unclassified	unclassified	unclassified	UL

1 Introduction

Robust grasping of previously unknown objects is an important capability for general-purpose robots. In order to grasp such an object, the robotic system must obtain relevant information from its sensors. Since the information obtained from the sensors and the execution of commands by the effectors are usually noisy processes, there is a need to develop algorithms that can perform in the presence of errors. The use of dextrous hands with many fingers can help in the solution of this problem. Since dextrous hands have more fingers than strictly necessary to grasp an object, the redundancy can be utilized to overcome errors in sensing and effecting. However, planning the position and force applied by each of the several fingers of a multi-fingered hand is a more complex problem than planning similar operations for a parallel-jaw gripper.

Fingertip grasps, or precision grasps, are normally planar [Cutkosky, 1985]. This suggests that they can be planned, and their execution can be monitored, using algorithms that don't require reconstruction of the three-dimensional world. Developing this idea, we have designed and built a system that plans and executes precision grasps using a single camera and four tactile sensors as the only source of information. In order to solve the problem of planning the position and force applied by each of the fingers, we cast it as a search problem and solve it using a genetic algorithm, which is a heuristic search strategy loosely inspired by biological evolution. Genetic algorithms are well-suited for this kind of problem because they are fast and are generally more robust than other techniques with respect to local minima.

The system we present in this paper takes an image of the object to be grasped as input, extracts contour and normal information about the object from the image, chooses the four points on the contour that will achieve the "best" grasp (according to an objective function that will be defined later), and then executes the grasp using visual and tactile information to close the control loop. Although our strategy is tailored to a four-fingered manipulator and a two-dimensional grasp strategy, it can easily be generalized to three dimensions and different manipulators.

The grasping system described in this paper is being used in conjunction with dextrous manipulation and adaptive calibration-free visual servoing systems to perform precision assembly tasks under visual guidance.

The organization of the remainder of the paper is as follows: section two describes related work, section three gives an overview of the system, sections four, five and six describe the image processing, grasp planning, and grasp execution modules of the system, respectively, section seven presents experimental results, and section eight presents conclusions.

2 Related Work

The appearance of dextrous manipulators, most notably the Salisbury hand [Salisbury, 1982] and the Utah/MIT hand [Jacobsen *et al.*, 1986; Jacobsen, 1984], in the early and mid-eighties motivated a great number of studies aimed at analyzing and understanding these mechanisms. These studies made almost exclusive use of classical deterministic techniques

of statics, kinematics, dynamics and control theory. In these analyses, it was assumed that complete models of the geometry of the hand and the grasped object were available. Uncertainty was considered later, but even then, the bound of the errors was assumed to be known.

Hanafusa and Asada [Hanafusa and Asada, 1982] derived necessary and sufficient conditions for achieving a stable grasp using a hand with elastic fingers. They showed that a grasp (which they called a prehension) was stable when the total energy stored in the elastic fingers was (locally) minimal with respect to small translations and rotations of the object being grasped. Salisbury investigated grasping models with and without friction using several types of contact between a finger and an object [Salisbury, 1982]. Mason and Salisbury [Mason and Salisbury, 1985] studied the mechanics of grasps for rigid body kinematics and Coulomb friction. They proposed a method for controlling the Salisbury hand that could produce small arbitrary motions and apply small arbitrary forces to a grasped object. Nguyen [Nguyen, 1988] showed that a polyhedral object can be stably grasped using three fingers assuming point contacts with friction. Lafferriere [Lafferriere, 1989] showed that for smooth objects (*i.e.*, objects with no vertices), four fingers always suffice to achieve a stable grasp, assuming contacts with arbitrarily small but non-zero friction. Montana [Montana, 1988] derived a set of equations that are a general description of the kinematics of contact between two rigid bodies. Using these equations, he analyzed the kinematics of grasping and derived a method of fine grip adjustment to obtain a grip with (locally) maximum stability for a two-fingered hand. Trinkle, Abel and Paul [Trinkle *et al.*, 1988] studied the mathematics of frictionless grasping. They presented a method for grasping a polygonal object with a two-dimensional hand composed of a palm and two hinged fingers. Mirtich and Canny [Mirtich and Canny, 1994] showed that, assuming rounded fingertips and contacts with friction, any 2-D object can be grasped with two fingers and any 3-D object can be grasped with three fingers. Ponce, Stam and Faverjon [Ponce *et al.*, 1993] presented algorithm for computing force-closure grasps (*i. e.* grasps where any movement of the object can be resisted by a contact force [Nguyen, 1988]) for piecewise-smooth curved 2-D objects. Coelho and Grupen [Coelho and Grupen, 1994] have studied the grasping problem, viewing it as a control composition problem. They presented force and moment controllers that achieve stable grasps on polygonal objects with four-fingered hands. Brost [Brost, 1988] presented a method for grasping a 2-D object that was insensitive to bounded errors in the location of the object.

When dextrous manipulators became widely available for experiments, it was realized that the complete models used for theoretical analyses were difficult to obtain in the real world, and that the worst-case analysis of uncertain models was unnecessarily restrictive. It was also difficult to determine the limits of uncertainty. This motivated researchers to search for methods that do not guarantee to find optimal grasps but use heuristics to obtain "good" grasps most of the time.

A large portion of the heuristic knowledge employed in these systems was obtained from observations made on human hands. Lyons [Lyons, 1985] defined the heuristic measures *firmness* and *precision* to evaluate the quality of a grasp quantitatively. Given the specifications of a task, Lyons's system chose, from a predefined set, the grasp characteristics that were best suited to the specifications, according to these heuristic measurements. Stans-

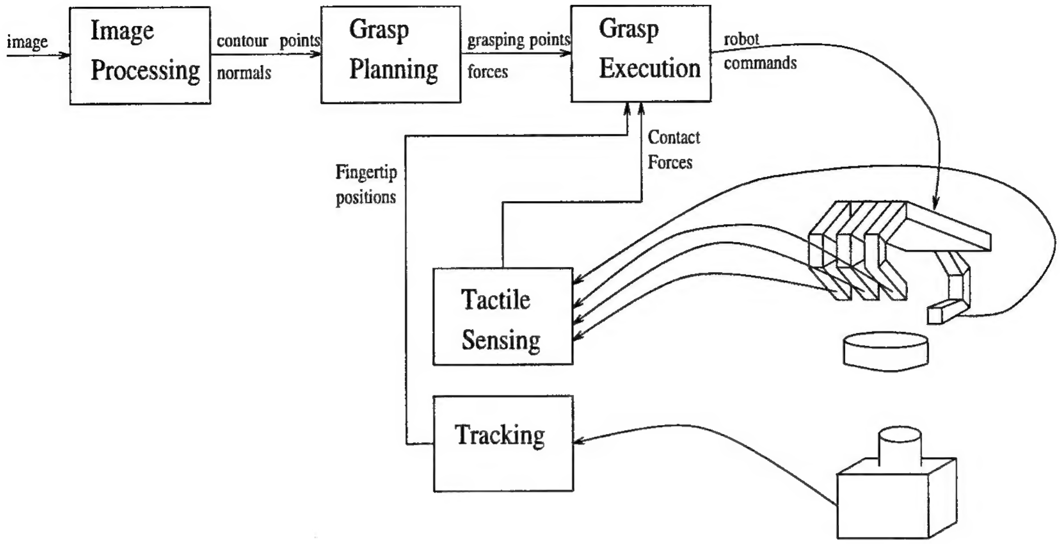

Figure 1: Schematic diagram of the vision-based grasping system

field [Stansfield, 1991] presented a knowledge-based system for deciding how to grasp several different unknown objects. The system performed full reconstruction of the object before grasping using a laser range-finder. From the reconstructed object it chose a grasping strategy using an expert system. Bekey *et al.* [Bekey *et al.*, 1993] developed a knowledge-based grasp planner for the University of Belgrade/USC robotic hand. Given a target object and a task, their planner chooses a grasp posture using four separate sources of heuristic knowledge, namely, knowledge about the robot hand, knowledge about the target object geometry, knowledge about the task, and knowledge about human grasping. This system is an advancement over previous ones in that it chooses grasp parameters as well as grasp types. Caselli *et al.* [Caselli *et al.*, 1993] presented a hybrid system for grasp synthesis that integrates symbolic and neural computations. The symbolic modules encode heuristic knowledge along with simple geometric reasoning, and the neural networks modules establish the more complex relationships between the geometric attributes of the object and the hand kinematics.

Genetic algorithms have been previously applied in robotics for trajectory optimization of redundant manipulators [Davidor, 1990], for motion planning [Ahuactzin *et al.*, 1992], and for generation of minimum distance paths for robot manipulators operating in cluttered environments [Solano and Jones, 1993].

3 Overview of the System

The general structure of the system is shown in figure 1. The input is an image showing the object to be grasped and the hand. Image analysis is performed to obtain a parametric representation of the object's contour and normal. Given this information, a genetic algorithm finds the position of the fingertips and the forces to be applied that achieve the best grasp according to a grasp quality metric. We consider grasp quality to be a function of

the object's shape and the fingers' ranges. After a grasp has been planned, the fingers are moved to preshape position using differential visual feedback, which allows the movements of the fingers to be specified in image coordinates without requiring an accurate calibration between the camera and the robot hand. From the preshape position, the fingers move slowly towards the center of the object, monitoring the magnitudes of the applied forces using tactile sensors located at the fingertips. When the measured forces are within a tolerance of those given by the planner, the fingers stop and the grasp is completed.

4 Image Analysis

The input to the image analysis module is a picture of the hand and the object, taken from below the transparent table where the object rests. This is roughly equivalent to using an eye-in-hand system. The tasks of the image analysis module are to segment the object to be grasped from the background, compute a parameterized representation of the object's contour and normals and find an estimate of the center of mass of the object. We use a simple flood-fill region-growing algorithm for segmentation. The initial seed is provided by the user by clicking with the mouse in the appropriate place in the image. After region-growing, we apply a median filter to remove small holes inside the boundary of the object. We compute an estimate of the center of mass by finding the mean of the indices of the pixels in the region. From the segmented image we do edge-finding and then edge-linking to obtain the object's contour. From the contour representation we obtain the normals by rotating the tangents by 90 degrees.

5 Grasp Planning

We view grasp planning as a process of optimizing a grasp quality metric. Intuitively, grasp quality is determined by how well the positions of the fingers and applied forces fulfill a set of object-dependent, task-dependent, and hand-dependent requirements. We define overall grasp quality as the product of five different functions, each attempting to capture an important aspect of grasp quality. For convenience, each of these functions will be designed to have values in the continuous interval $[0, 1]$, where a value of 1 corresponds to a perfect grasp.

We will restrict our analysis to a four-fingered hand, such as the Utah/MIT hand, with a thumb, denoted as finger zero, and three opposing fingers. We will assume that the z components of the positions of the fingertips are fixed and equal. The goal of the grasp planning procedure is to find the $\langle x, y \rangle$ positions of the fingertips and the force magnitudes that achieve a good grasp, according to an optimality criterion.

In our grasp strategy, each finger applies a force to the object in the direction of its center of mass. This ensures that the sum of moments on the object is zero and also simplifies the calculation of the force magnitudes. We assume point contacts with friction, although we will ultimately define a function that explicitly disallows grasps that are not achievable with real (non point) fingers. Figure 2 illustrates our strategy. The contour of the object, parameterized by arc length, is represented by $\alpha(s)$. $\alpha(s_0), \ldots, \alpha(s_3)$ are the

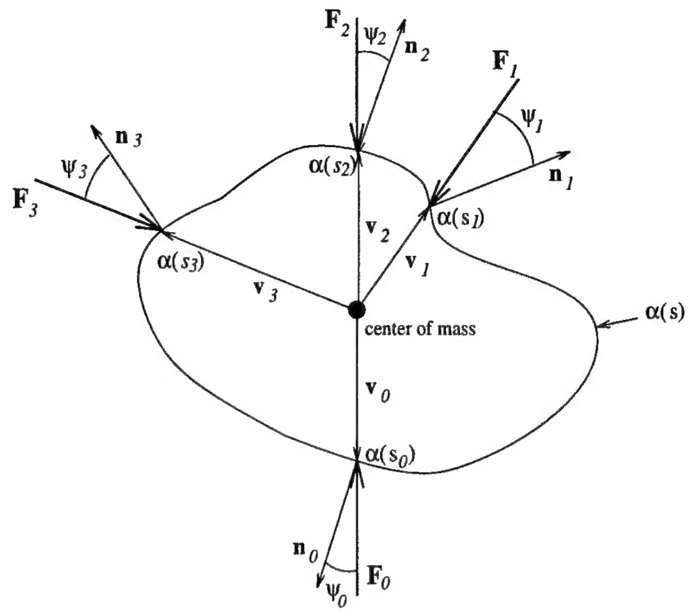

Figure 2: Grasping strategy

coordinates of the fingertip to object contact points. $\mathbf{n}_0, \ldots, \mathbf{n}_3$ are the normals at those points. $\mathbf{v}_0, \ldots, \mathbf{v}_3$ are the vectors going from the center of mass of the object to the contact points $\alpha(s_0), \ldots, \alpha(s_3)$.

We consider the following grasp requirements and the corresponding functions:

1. The object must be in static equilibrium and the computed force for each finger must be physically realizable. For the object to be in equilibrium, the sum of forces and the sum of moments on the object must be equal to zero.

$$\sum_i \mathbf{F}_i = \mathbf{0}$$

and

$$\sum_i \mathbf{F}_i \times \mathbf{v}_i = \mathbf{0}$$

Since $\mathbf{F}_i$ and $\mathbf{v}_i$ are colinear, our grasp strategy guarantees that the sum of moments will always be zero.

The equation for the sum of forces can be rewritten as:

$$\sum_i \mathbf{F}_i = -\sum_i m_i \frac{\mathbf{v}_i}{|\mathbf{v}_i|} = \mathbf{0}$$

where m_i is the magnitude of the force applied by finger i. We can arbitrarily scale the magnitudes of a system of forces in equilibrium and still preserve equilibrium, thus we only need to compute the force magnitudes up to a scale factor. Later we

can scale the magnitudes appropriately depending on the object's estimated weight and the task requirements. In order to simplify the problem, we assume that two adjacent fingers can act together as a "virtual finger" [Iberall, 1987]. Specifically, we assume that the forces applied by fingers two and three (middle and ring fingers, in anthropomorphic terms) are equal. Thus, setting $m_0 = 1$ and $m_2 = m_3$, we are left with two equations with two unknowns and we can solve for m_1 and m_2. We could relax the $m_2 = m_3$ assumption and instead solve the system of equations using a pseudo-inverse-based method, but this is computationally expensive and would make he function unsuitable for repeated evaluation in real time. If $m_1 < 0$ or $m_2 < 0$, one or more of the fingers is required to exert a tension force on the object, which is impossible. Also, m_1 and m_2 have to be within the physical limits of the robot. A good heuristic, based on observations of human hands, is to prefer grasps in which the magnitude of the force applied by the thumb is twice as large as the forces applied by the other fingers.

Using these heuristics, and the requirements that each grasp function must fall in the $[0, 1]$ interval, with a value of 1 representing a perfect grasp, we define the grasp function $g_1(\langle s_0, s_1, s_2, s_3 \rangle)$ as:

$$g_1(\mathbf{s}) = \begin{cases} e^{-\sum_{i=1}^{3} abs(m_i - \frac{1}{2})} & \text{if } m_1 \geq 0 \text{ and } m_2 \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

2. All forces must be within the cone of friction. This means that the ratio of tangential to normal force at any contact must not exceed the friction coefficient of the contact.

   Since the friction coefficient is unknown, we restate this requirement as a minimization of the friction coefficient required to achieve the grasp. Minimizing the required friction coefficient is equivalent to maximizing, for every contact, the ratio of normal to total force applied. For contact i, this ratio is simply $\cos(\psi_i)$.

   Thus, our second grasp quality function is defined as the worst normal to total force ratio of the contact points.

$$g_2(\mathbf{s}) = \min_{i \in \{0,\ldots,3\}} (\cos(\psi_i))$$

3. The grasp must be stable with respect to small movements of the fingertips along the boundary of the object. A good heuristic to achieve this, which has also been used in [Murphy *et al.*, 1993], is to minimize the sum of distances from each contact point to the center of mass of the object.

   We need a function that minimizes $\sum_i |\mathbf{v}_i|$. Let $L = \max(|\alpha(s) - \mathbf{c}|)$, let $l = \min(|\alpha(s) - \mathbf{c}|)$, where $\mathbf{c}$ is the center of mass. The upper and lower bounds on $\sum_i |\mathbf{v_i}|$ are $4L$ and $4l$, respectively. We then define our next grasp function as:

$$g_3(\mathbf{s}) = \frac{4L - \sum_i |\mathbf{v}_i| + \epsilon}{4(L - l) + \epsilon}$$

Where ϵ is a very small constant whose only use is avoiding division by zero in some degenerate cases.

4. In order to maximize dexterity after grasping, the fingers should be positioned as far away as possible from their range limits. This is approximated by trying to minimize the sum of distances from each proposed contact point to the center of the range of the corresponding finger. We will also disallow grasps that require any of the fingertips to go outside of its range.

   Let $\mathbf{d}_i$ be the coordinates of the center of the range of finger i. Let r_i be the range of the finger in the direction of the proposed contact point $\alpha(s_i)$. Clearly, if $|\mathbf{d}_i - \alpha(s_i)| > r_i$, the grasp is not realizable, and the amount of dexterity left after grasping will increase as $|\mathbf{d}_i - \alpha(s_i)|$ decreases.

$$g_4(\mathbf{s}) = \begin{cases} avg(\frac{|\mathbf{d}_i - \alpha(s_i)|}{r_i}) & \text{if } \min(\frac{|\mathbf{d}_i - \alpha(s_i)|}{r_i}) > 0 \\ 0 & \text{otherwise} \end{cases}$$

5. Although we are assuming point contacts with friction, the fingertips are not points. Therefore we must not allow grasps in which the distance between two of the fingers is less than the sum of their radii.

$$g_5(\mathbf{s}) = \begin{cases} 0 & \text{if } \min_{i,j} |\alpha(s_i) - \alpha(s_j)| < (r_i + r_j), \text{ for } i, j \in \{0, \ldots, 3\}, i \neq j \\ 1 & \text{otherwise} \end{cases}$$

Finally, the overall quality metric Q is given by the product of the grasp quality functions.

$$Q(\mathbf{s}) = \prod_{j=1}^{5} g_j(\mathbf{s})$$

We use product instead of other alternatives such as weighted sum because we want to maximize all the grasping functions simultaneously. If a proposed grasp does not fulfill one of the requirements, it is not a good grasp, even if it satisfies all the other requirements very well.

We can now solve the grasp planning problem by searching for the values of $\langle s_0, \ldots, s_3 \rangle$ that maximize $Q(\mathbf{s})$.

Several previous papers have posed the grasp planning problem as an optimization problem. Hanafusa and Asada [Hanafusa and Asada, 1982] defined an optimality function for a three-degree-of-freedom hand with elastic fingers based on the potential energy stored by the fingers. They also proved that local minima in their potential function corresponded to stable grasps. Jameson and Leifer [Jameson and Leifer, 1987] found stable grasps by maximizing the distance from the configuration to the nearest unstable configuration. They used a modified Newton method. Woelfl and Pfeiffer [Woelfl and Pfeiffer, 1994] used the Successive Quadratic Programming technique. Seitz and Kraft [Seitz and Kraft, 1994] performed an exhaustive search. Bendiksen and Hager [Bendiksen and Hager, 1994] used the simplex method for a one-dimensional search for the optimal parameter of their grasping system.

5.1 A genetic algorithm for finding grasping points

Since our grasp quality metric is non-differentiable globally, and may also exhibit local minima, classical optimization techniques may be difficult to apply. We solve this optimization problem using a genetic algorithm, modified to deal with a continuous parameter space. The advantages of this method are that it does not require differentiability of the objective function, it is usually fast, and, since the search is not limited to a local neighborhood, it is robust with respect to local minima. The main disadvantage is that it is not guaranteed to find the optimal solution in a finite amount of time. But, as our experiments indicate, it usually finds a good enough solution in a reasonable amount of time.

Genetic algorithms are an optimization method loosely based on biological evolution [Holland, 1975], [Goldberg, 1989]. Unlike most optimization methods, which typically modify a single candidate solution iteratively, genetic algorithms work with a set of candidate solutions, called a *population*. Candidate solutions are encoded as strings, normally, but not necessarily, over a binary alphabet. During each iteration of the algorithm, a new population of strings is obtained from the old one using operations that mimic natural selection. Genetic algorithms generally use three basic operators: replication, in which a string in the old population is copied to the new one, crossover, in which portions of two strings are interchanged, and mutation, in which one or more pieces of a string are randomly changed. Natural selection is mimicked by giving a "good" string a greater probability of participating in the creation of the new population. The measure of the "goodness" of a string is called its *fitness*. An individual is chosen to participate in the creation of the next generation with a probability proportional to its fitness.

The algorithm we use is a variant of traditional binary-encoded genetic algorithms and is based on an algorithm originally described by Grossman and Davidor [Grossman and Davidor, 1992]. The use of representations that are more expressive than binary encoding has been advocated by Antonisse [Antonisse, 1989] and by Janikow and Michalewicz [Janikow and Michalewicz, 1991].

The input to the algorithm is the list of points that form the boundary of the object, as obtained by the image analysis module. The genetic algorithm finds four points on the boundary of the object that yield a good grasp according to the grasp quality function.

In the design of a genetic algorithm, three main issues generally arise. The first is how to efficiently encode candidate solutions as strings. This includes choosing a suitable alphabet for the strings in the population, or perhaps a set of alphabets for different portions of the strings. The second is how to evaluate the fitness of the strings in a way that will yield the fastest convergence to a correct solution. The third is choosing genetic operators that are appropriate to the problem and the representation used. In the following paragraphs we will discuss the way we deal with these issues in our system.

Encoding of Individuals Let $\alpha : [0,1] \rightarrow \Re^2$ be the parameterized representation of the object's contour obtained by the image analysis module. An individual $\mathbf{s} = \langle s_0, s_1, s_2, s_3 \rangle \in [0,1]^4$ is a quadruple that encodes the grasp with fingertip positions $\langle \alpha(s_0), \alpha(s_1), \alpha(s_2), \alpha(s_3) \rangle$.

Fitness Evaluation The grasp quality function defined in the previous section was purposedly designed to be suitable for use as a fitness function. The fitness of an individual is thus equal to the grasp quality for the grasp encoded by that individual. That is, $f(\mathbf{s}) = Q(\mathbf{s})$.

Genetic Operators We use the traditional crossover operator and the mutation operator, modified to work for the case of a real-number representation. We also use the interpolation and extrapolation operators, originally proposed in [Grossman and Davidor, 1992].

Crossover Given two individuals $\mathbf{s} = \langle s_0^I, s_1^I, s_2^I, s_3^I \rangle$ and $\mathbf{s}^J = \langle s_0^J, s_1^J, s_2^J, s_3^J, \rangle$, we randomly choose a crossover point $p \in \{0, 1, 2\}$ and obtain two new individuals $\mathbf{s}^K = \langle s_0^I, \ldots, s_p^I, s_{p+1}^J, \ldots, s_3^J \rangle$ and $\mathbf{s}^L = \langle s_0^J, \ldots, s_p^J, s_{p+1}^I, \ldots, s_3^I \rangle$.

Mutation A mutation in our algorithm is the addition to an individual of a normally distributed random vector with zero mean.

$$\mathbf{s}^{new} = \mathbf{s}^{old} + \langle r_0, r_1, r_2, r_3 \rangle$$

where $r_0, \ldots, r_3$ are scalars randomly chosen from the normal distribution. After performing the operation the values of $\langle s_0, s_1, s_2, s_3 \rangle$ are wrapped around in case they fall out of the $[1, n]$ range .

Interpolation A new individual is obtained by a weighted average of its two parents. In our implementation we use equal weights for the parents, but some other schemes such as varying relative weights randomly or assigning a higher weight to the fitter parent can be used.

$$\mathbf{s}^{new} = \frac{1}{2}\mathbf{s}_\mathbf{x}^{old} + \frac{1}{2}\mathbf{s}_\mathbf{y}^{old}$$

Extrapolation If we have two individuals $\mathbf{s_x}$ and $\mathbf{s_y}$ with fitnesses $f(\mathbf{s_x})$ and $f(\mathbf{s_y})$, with $f(\mathbf{s_x}) > f(\mathbf{s_y})$, a good heuristic for finding an individual $\mathbf{s_z}$ that is fitter than $\mathbf{s_x}$ is to extrapolate the values of $\mathbf{s_x}$ and $\mathbf{s_y}$ in the direction of increased fitness, as illustrated in figure 3.

$$\mathbf{s_z} = \mathbf{s_x} + \beta(\mathbf{s_x} - \mathbf{s_y})$$

where $\beta > 0$ is a constant.

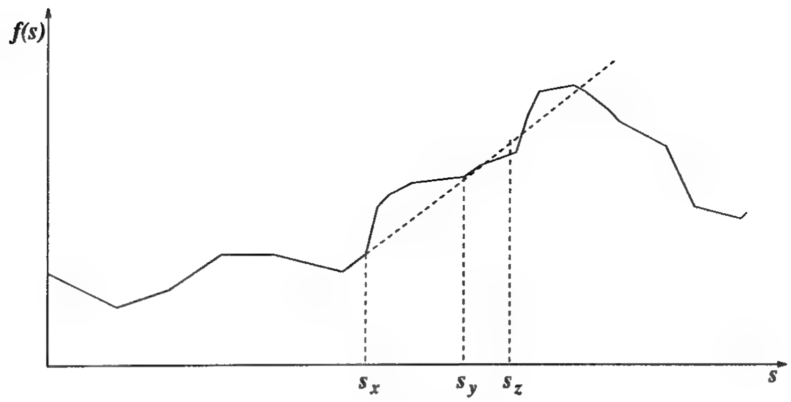

Figure 3: Extrapolation operator in one dimension

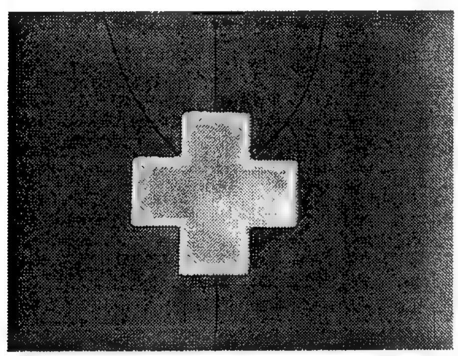

Figure 4: Planned grasping points and fingertip trajectories for the cross

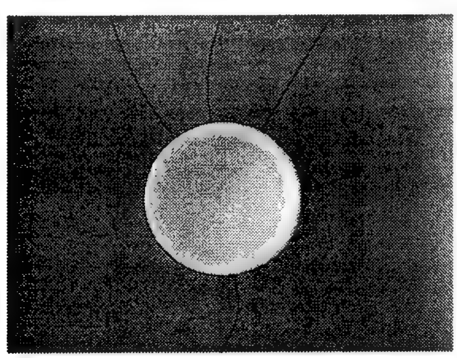

Figure 5: Planned grasping points and fingertip trajectories for the circle

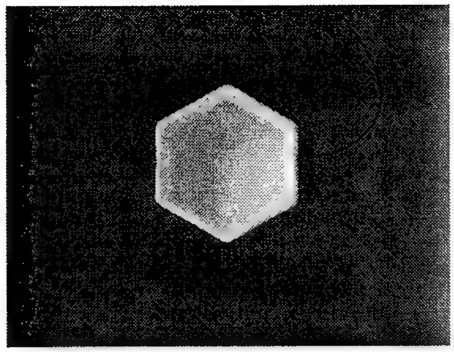

Figure 6: Planned grasping points and fingertip trajectories for the hexagon

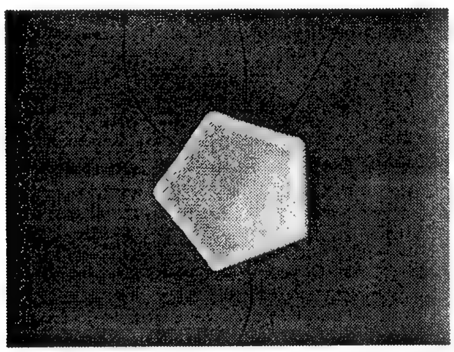

Figure 7: Planned grasping points and fingertip trajectories for the pentagon

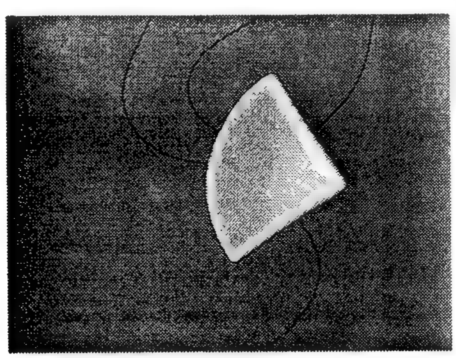

Figure 8: Planned grasping points and fingertip trajectories for the pie slice

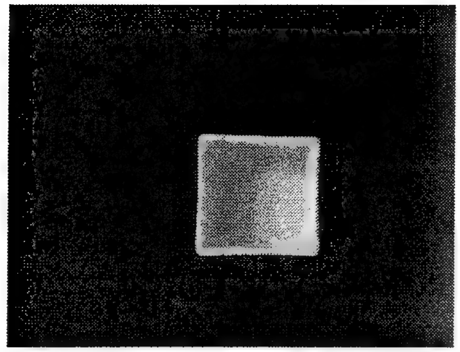

Figure 9: Planned grasping points and fingertip trajectories for the square

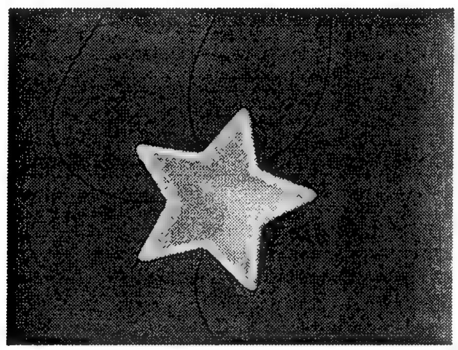

Figure 10: Planned grasping points and fingertip trajectories for the star

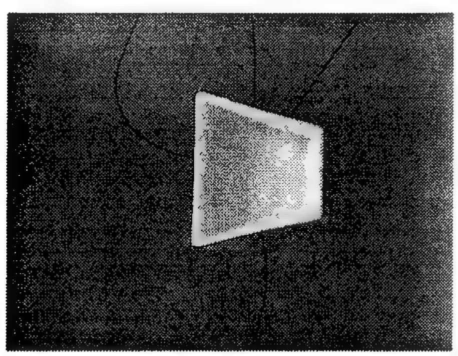

Figure 11: Planned grasping points and fingertip trajectories for the trapezoid

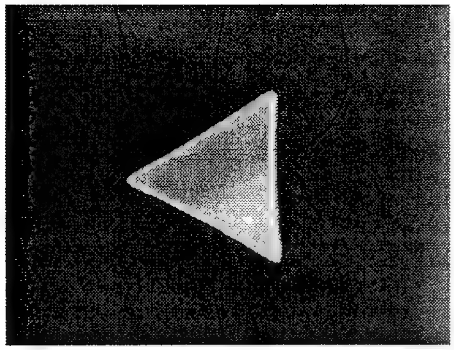

Figure 12: Planned grasping points and fingertip trajectories for the triangle

5.2 Results

After the grasping points have been selected, a smooth trajectory from the initial to the contact position is planned for each finger. The goal is for each fingertip to approach the object in the same direction from which the force will be applied. Figures 4 to 12 show the contact points and the corresponding trajectories found in a typical run of the program and the corresponding trajectories. It was assumed that initially the thumb was at the center of the bottom edge of the image, the index near the top right corner, the middle finger at the center of the top edge, and the ring finger near the top left corner, as shown in the figures. Additional results showing the functioning of the whole system, including the grasp execution module, will be presented in section 7.

6 Grasp Execution

After the grasping points have been selected by the grasp planner, the grasp execution system has to move the fingertips to the desired positions. The fingers are taken to the desired positions using differential visual feedback [Feddema *et al.*, 1992; Wijesoma *et al.*, 1993; Martin Jägersand, 1994], a calibration-free scheme for specifying and executing tasks in visual space.

Once the fingers are in their desired position we switch to force-control mode. Using tactile sensors we move the fingers toward the center of the object until the pressures at the fingers are within a threshold of the forces computed by the grasp planner.

6.1 Differential Visual Feedback

Differential visual feedback allows a system to operate without a prior calibrated model of the relationship between the camera and the 3-dimensional scene. The basic idea is that even if the exact calibration between the image space and the robot coordinate system is unknown, an approximate Jacobian matrix mapping changes in robot space to changes in image space can be obtained and used iteratively to guide the robot to attain a goal specified in image space.

Let $\mathbf{x}^R = \langle x^R, y^R \rangle$ be the coordinates of the finger in the hand's reference frame. Let $\mathbf{x}^I = \langle x^I, y^I \rangle$ be the coordinates in the finger in image space. We are interested in finding the Jacobian $\mathbf{J}$, a 2×2 matrix that relates changes in $\mathbf{x}^R$ to changes in $\mathbf{x}^I$.

$$\begin{bmatrix} \Delta x^I \\ \Delta y^I \end{bmatrix} = \begin{bmatrix} \frac{\partial x^I}{\partial x^R} & \frac{\partial x^I}{\partial y^R} \\ \frac{\partial y^I}{\partial x^R} & \frac{\partial y^I}{\partial y^R} \end{bmatrix} \begin{bmatrix} \Delta x^R \\ \Delta y^R \end{bmatrix}$$

Let $\langle x_0^I, y_0^I \rangle$ be the start coordinates of a finger, in image coordinates. We perform a calibration movement $\langle 1, 0 \rangle$ that yields a change in the position of the finger of $\langle \Delta x_{dx}^I, \Delta y_{dx}^I \rangle$ in the image. We then perform another calibration movement $\langle 0, 1 \rangle$ obtaining a displacement of $\langle \Delta x_{dy}^I, \Delta y_{dy}^I \rangle$ in the image. From these measurements, an accurate approximation to the Jacobian can be obtained.

$$\begin{bmatrix} \Delta x^I \\ \Delta y^I \end{bmatrix} = \begin{bmatrix} \Delta x_{dx}^I & \Delta x_{dy}^I \\ \Delta y_{dx}^I & \Delta y_{dy}^I \end{bmatrix} \begin{bmatrix} \Delta x^R \\ \Delta y^R \end{bmatrix}$$

We want to obtain the robot commands corresponding to a visual goal computed by the system, so the equation that we really need is:

$$\begin{bmatrix} \Delta x^R \\ \Delta y^R \end{bmatrix} = \begin{bmatrix} \Delta x_{dx}^I & \Delta x_{dy}^I \\ \Delta y_{dx}^I & \Delta y_{dy}^I \end{bmatrix}^{-1} \begin{bmatrix} \Delta x^I \\ \Delta y^I \end{bmatrix}$$

For this application we keep $\mathbf{J}$ constant, although it is possible to update it as the task progresses [Martin Jägersand, 1994]. This is useful when the mapping between the image space and the robot frame is changing, such as when robot commands are changes in joint values instead of changes in the position of the end-effector.

6.2 Tactile Sensing and Force Control

While performing the fingertip position adjustments we monitor the tactile sensors located at each fingertip. When the reading in a fingertip exceeds a predetermined small threshold we assume that the finger has come in contact with the object. At this point visual control of the finger stops. We continue this process until all of the fingers come into contact with the object, then we switch to a simple force controller. Each fingertip moves towards the center of grasp until the pressure read by the sensors is within a tolerance of the force computed by the planner.

7 Experimental Results

In the experiments, we successfully grasped each of the objects shown in figures 4 to 12 using the four-fingered Utah/MIT hand. Figure 13 shows an image of the hand and an object, prior to grasping. The image was taken from beneath the transparent table where the object rested. The figure also shows the contour of the object, obtained by the image

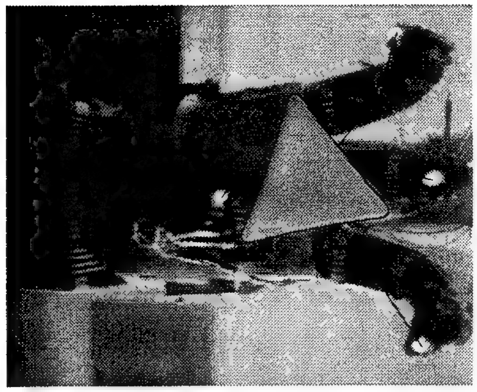

Figure 13: Initial position of the hand, segmented object contour, and planned fingertip trajectories

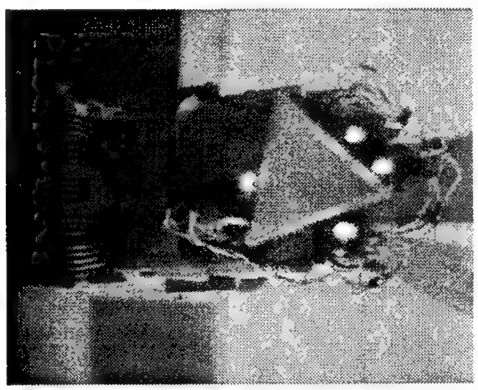

Figure 14: Final fingertip positions after executing the grasp

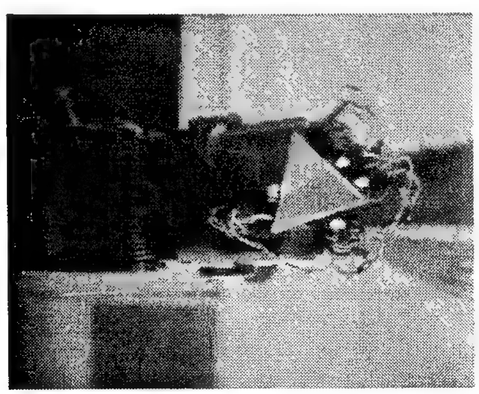

Figure 15: Object is picked-up

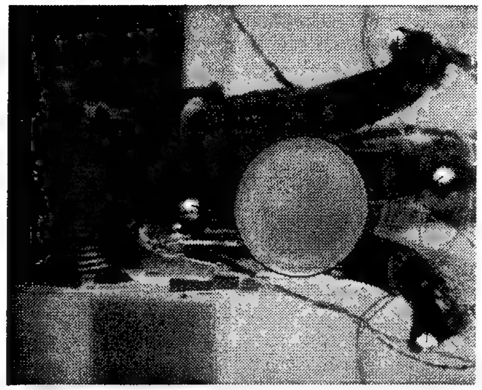

Figure 16: Initial position of the hand, segmented object contour, and planned fingertip trajectories

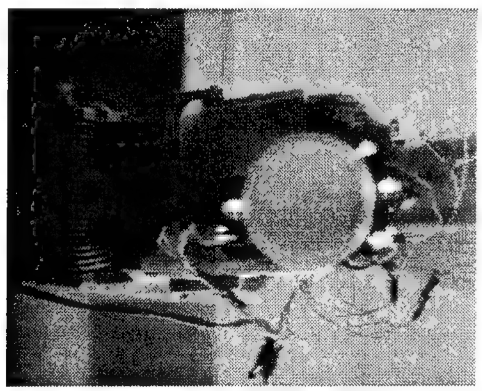

Figure 17: Final fingertip positions after executing the grasp

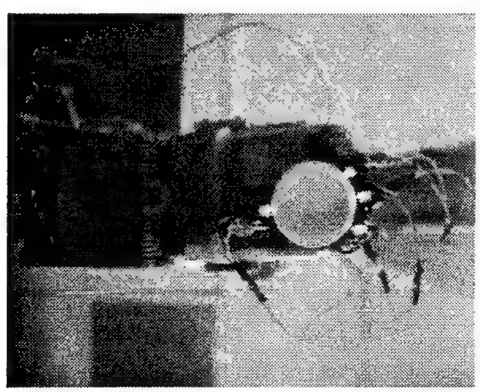

Figure 18: Object is picked-up

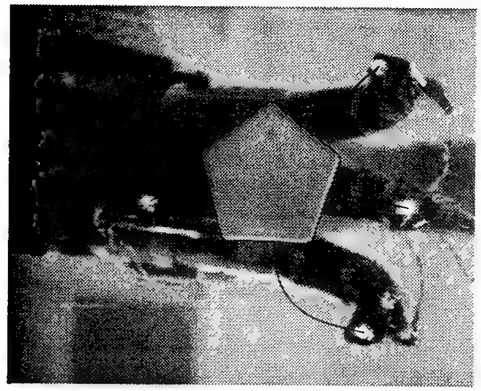

Figure 19: Initial position of the hand, segmented object contour, and planned fingertip trajectories

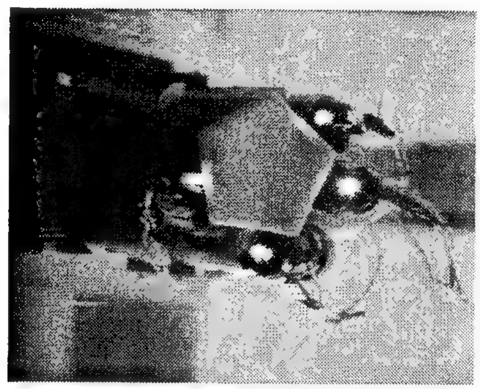

Figure 20: Final fingertip positions after executing the grasp

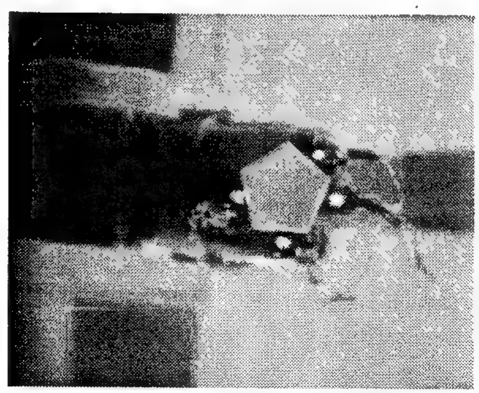

Figure 21: Object is picked-up

	Tactile-based Grasp		Vision-based Grasp	
	Grasping	Manipulation	Grasping	Manipulation
Hexagon	0	4	0	0
Triangle	0	10	0	2

Table 1: Number of failures in ten trials using the hexagon and the triangle

analysis module, and the planned trajectories of the fingers given by the planner. We attached LEDs to the fingertips to facilitate tracking. Figure 14 shows the object and the hand after grasping, figure 15 shows the hand and the object after the object has been picked-up. Similarly, figures 16 to 21 show the grasping of two other objects from the collection. For all experiments, the genetic algorithm was run for 100 generations, with a population size of 200. After the predetermined number of generations were run, the algorithm output the best individual in the final population.

7.1 Comparison to Other Approaches

In order to evaluate the performance of the algorithm relative to other techniques, we also implemented a simple tactile-based grasping method. In this method, the hand starts in a fixed preshape position and the fingers move simultaneously to the center of the preshape. When the forces read from the tactile sensors exceed a small threshold, the fingers stop, meaning that the object has been contacted. From this position each finger moves a fixed distance toward the center of grasp, in order to apply some force to the object. We have previously used this strategy with some success as the first step for manipulation with the Utah/MIT hand.

We made some preliminary experiments for the comparison between the two approaches. It was expected that for easy-to-grasp objects, such as the circle or the hexagon, both strategies would be equally effective, but that for more difficult objects, such as the triangle, the more complex vision-based strategy would perform better. It was also expected that the vision-based strategy would increase manipulability, since the grasp quality metric explicitly attempts to maximize the range of motion each finger has after performing the grasp. To test this hypotheses, we performed some experiments using as target objects the hexagon and triangle, which are representative instances of easy and hard to grasp objects, respectively. In order to test the suitability of a grasp, the hand needs to pick up the object and then to perform some manipulations with it. Each trial consists of two phases: the grasping phase, in which the object is grasped and picked-up, and the manipulation phase, in which we apply a series of quick translations and rotations to the object. The method for performing these manipulations is described in detail in [Fuentes, 1994]. We performed 40 trials; 10 for each object-strategy pair. A trial was classified as a grasping failure if the object was not successfully picked-up. A trial was classified as a manipulation failure if the object was dropped while being manipulated. Otherwise the trial was classified as a success. The comparative results of both strategies are summarized in Table 1.

Although the tests are far from exhaustive, it is possible to draw some conclusions about the relative performance of the two strategies. The experiments suggest that if the only goal is to grasp the object securely, both strategies perform satisfactorily, and thus the overhead created by the use of the vision-based system might be unjustified. However, if the goal is not just to grasp the object, but also to manipulate it, the vision-based strategy easily outperforms the tactile-based strategy.

8 Conclusions

We have presented a system for planning and executing planar grasps of unknown objects using visual and tactile information. We posed grasp planning as a search problem and solved it using a genetic algorithm. Grasping was performed using differential visual feedback, a method for uncalibrated visual servoing, and tactile sensing to monitor the forces applied to the object. Our system performs very well with all the objects we tried and runs in a reasonable amount of time. It was shown that for tasks requiring manipulation after grasping, our method easily outperforms a simpler tactile-based strategy.

References

[Ahuactzin *et al.*, 1992] Juan Manuel Ahuactzin, El-Ghazali Talbi, Pierre Bessiere, and Emmanuel Mazer, "Using genetic algorithms for robot motion planning," in *ECAI 92. 10th European Conference on Artificial Intelligence Proceedings*, pages 672–675, 1992.

[Antonisse, 1989] J. Antonisse, "A New Interpretation of the Schema Notation that Overturns the Binary Encoding Constraint," in *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 86–91. Morgan Kaufmann, 1989.

[Bekey *et al.*, 1993] G. A. Bekey, H. L. Liu, R. Tomovic, and W. J Karplus, "Knowledge-Based Control of Grasping in Robot Hands Using Heuristics from Human Motor Skills," *IEEE Transactions on Robotics and Automation*, 9(6):709–722, December 1993.

[Bendiksen and Hager, 1994] Aage Bendiksen and Greg Hager, "A Vision-Based Grasping System for Unfamiliar Objects," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2844–2849, San Diego, California, 1994.

[Brost, 1988] Randy Brost, "Automatic Grasp Planning in the Presence of Uncertainty," *The International Journal of Robotics Research*, 7(1):3–17, February 1988.

[Caselli *et al.*, 1993] Stefano Caselli, Eugenio Faldella, Bruno Fringuelli, and Francesco Zanichelli, "A Hybrid System for Knowledge-Based Synthesis of Robot Grasps," in *IEEE/RSJ International Conference on Intelligent Robots an Systems*, pages 1575–1581, Yokohama, Japan, 1993.

[Coelho and Grupen, 1994] Jefferson A. Coelho and Roderic A. Grupen, "Effective Multi-fingered Grasp Synthesis," Technical Report 94-12, Laboratory for Perceptual Robotics, Department of Computer and Information Science, University of Massachusetts at Amherst, Amherst, Massachusetts, 1994.

[Cutkosky, 1985] M. R. Cutkosky, *Robotic Grasping and Fine Manipulation*, Kluwer Academic Publishers, 1985.

[Davidor, 1990] Y. Davidor, "A genetic algorithm applied to robot path-planning," Technical Report CS90-01, The Weizmann Institute of Science - Department of Applied Mathematics and Computer Science, 76100 Rehovot, Israel, January 1990.

[Feddema *et al.*, 1992] J. T. Feddema, C. S. G. Lee, and O. R. Mitchell, "Model-Based Visual Control for a Hand-Eye Coordinated Robotic System," *Computer*, pages 21–31, August 1992.

[Fuentes, 1994] Olac Fuentes, "Dexterous Manipulation Without Object Models". Area Paper. Computer Science Department, University of Rochester, April 1994.

[Goldberg, 1989] David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[Grossman and Davidor, 1992] T. Grossman and Y. Davidor, "An Investigation of a Genetic Algorithm in Continuous Parameter Space," Technical Report CS92-20, The Weizmann Institute of Science, Department of Applied Mathematics and Computer Science, Revhovot, Israel, October 1992.

[Hanafusa and Asada, 1982] H. Hanafusa and H. Asada, "Stable Prehension by a Robot Hand with Elastic Fingers," in M. Brady, editor, *Robot Motion*, pages 323–335. MIT Press, Cambridge, MA, 1982.

[Holland, 1975] John Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, MI, 1975.

[Iberall, 1987] T. Iberall, "The nature of human prehension: Three dextrous hands in one," in *Proceedings of the 1987 IEEE International Conference on Robotics and Automation*, pages 396–401, Raleigh, North Carolina, 1987.

[Jacobsen *et al.*, 1986] Steve Jacobsen, E. Iversen, D. Knutti, R. Johnson, and K. Bigger, "Design of the Utah/MIT Dextrous Hand," in *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, pages 96–102, 1986.

[Jacobsen, 1984] Steve C. Jacobsen, "The Utah/MIT hand: Work in Progress," *IEEE Journal of Robotics and Automation*, 3(4), 1984.

[Jameson and Leifer, 1987] John W. Jameson and Larry J. Leifer, "Automatic grasping: An Optimization Approach," *IEEE Transactions on Systems, Man and Cybernetics*, 17(5):806–813, September/October 1987.

[Janikow and Michalewicz, 1991] C. Z. Janikow and Z. Michalewicz, "An Experimental Comparison of binary and Floating Point Representations in Genetic Algorithms," in *Proceedings of the 4th International Conference on Genetic Algorithms*, pages 31–36. Morgan Kaufmann, 1991.

[Lafferriere, 1989] G. Lafferriere, "Multifinger Grasps for Fine Manipulation of Smooth Objects," Technical Report 457, New York University, Courant Institute of Mathematical Sciences, New York, New York, July 1989.

[Lyons, 1985] Damian Lyons, "A Simple Set of Grasps for a Dextrous Hand," in *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, pages 544–548, St. Louis, Missouri, 1985.

[Martin Jägersand, 1994] Martin Jägersand, "Perception level control for uncalibrated hand-eye coordination and motor actions". Area Paper. Computer Science Department, University of Rochester, April 1994.

[Mason and Salisbury, 1985] Matthew Mason and Kenneth Salisbury, *Robot Hands and the Mechanics of Manipulation*, The MIT Press, Cambridge, Massachusetts, 1985.

[Mirtich and Canny, 1994] Brian Mirtich and John Canny, "Easily Computable Optimum Grasps in 2-D and 3-D," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 739–747, San Diego, California, 1994.

[Montana, 1988] David J. Montana, "The Kinematics of Contact and Grasp," *The International Journal of Robotics Research*, 7(3):17–32, June 1988.

[Murphy *et al.*, 1993] T. G. Murphy, D. Lyons, and A. J. Hendricks, "Stable Grasping with a Multi-Fingered Robot Hand: A Behavior-Based Approach," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 867–874, Yokohama, Japan, 1993.

[Nguyen, 1988] Van-Duc Nguyen, "Constructing Force-Closure Grasps," *The International Journal of Robotics Research*, 7(3):3–16, June 1988.

[Ponce *et al.*, 1993] Jean Ponce, Darrell Stam, and Bernard Faverjon, "On Computing Two-Finger Force-Closure Grasps of Curved 2-D Objects," *The International Journal of Robotics Research*, 12(3):263–273, June 1993.

[Salisbury, 1982] K. Salisbury, *Kinematic and Force Analysis of Articulated Hands*, PhD thesis, Stanford University., Department of Mechanical Engineering, Stanford, California, 1982.

[Seitz and Kraft, 1994] Matthias Seitz and Jochen Kraft, "Some Approaches to context Based Grasp Planning for a Multi-fingered Gripper," in *Proceedings of the 1994 IEEE/RSJ International Conference on Intelligent Robots an Systems*, pages 360–365, Munich, Germany, 1994.

[Solano and Jones, 1993] J. Solano and D. I. Jones, "Generation of collision-free paths, a genetic approach," in *IEEE Colloquium on Genetic Algorithms for Control Systems Engineering*, pages 5/1–5/6, 1993.

[Stansfield, 1991] S. Stansfield, "Robotic Grasping of Unknown Objects: A Knowledge-based Approach," *International Journal of Robotics Research*, 1991.

[Trinkle *et al.*, 1988] Jeffrey C. Trinkle, Jacob M. Abel, and Richard P. Paul, "An Investigation of Frictionless Enveloping Grasping in the Plane," *The International Journal of Robotics Research*, 7(3):33–51, June 1988.

[Wijesoma *et al.*, 1993] S. W. Wijesoma, D. F. H. Wolfe, and R. J. Richards, "Eye-to-Hand Coordination for Vision-Guided Robot Control Applications," *The International Journal of Robotics Research*, 12(1):65–78, February 1993.

[Woelfl and Pfeiffer, 1994] Kurt Woelfl and Friedrich Pfeiffer, "Grasp Strategies for a Dextrous Robotic Hand," in *Proceedings of the 1994 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 366–373, Munich, Germany, 1994.